

**MINOR IN ENERGY PROJECT:  
DEVELOPMENT OF THE DYNAMIC POWER SYSTEM  
SECURITY AND RELIABILITY ASSESSMENT TOOLS**

**Lausanne  
Autumn 2011**

**PERRAUDIN Nathanaël**

supervised by  
**CHERKAoui Rachid**  
and  
**BIZUMIC Lazar**  
PWRS - EPFL



## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Definitions . . . . .	1
<b>2</b>	<b>General method</b>	<b>2</b>
2.1	The Monte Carlo simulations . . . . .	2
2.1.1	List of all possible scenarios . . . . .	2
2.1.2	Calculating the probability of scenarios . . . . .	3
2.2	The Poisson law . . . . .	3
2.3	Chosing one scenario . . . . .	3
2.3.1	The logical strategy . . . . .	4
2.3.2	Another scenario strategy . . . . .	4
2.3.3	Correlation between events . . . . .	5
2.4	Probability of an event in a steady and unoverloaded state . . . . .	5
2.4.1	The values used . . . . .	5
2.4.2	Finding the length of the lines . . . . .	5
2.4.3	Calculating the $pe_i$ values . . . . .	5
<b>3</b>	<b>The simulation</b>	<b>6</b>
3.1	Testing the grid . . . . .	7
3.1.1	Lines . . . . .	7
3.1.2	The generators . . . . .	7
3.2	End of simulation . . . . .	8
3.2.1	Regulation strategy . . . . .	8
<b>4</b>	<b>Calculation of the average loss of load</b>	<b>8</b>
4.1	Traditional strategy . . . . .	8
4.2	Optimised strategy . . . . .	9
4.3	Annualized expected loss of load . . . . .	9
4.4	The probability density function (PDF) . . . . .	9
<b>5</b>	<b>Gaussian mixture method (GMM)</b>	<b>9</b>
<b>6</b>	<b>Results</b>	<b>11</b>
6.1	Comparison between traditional and optimised strategy: . . . . .	11
6.2	Comparison between correlated and uncorrelated scenarios: . . . . .	11
6.3	Influence of parameter $\alpha$ : . . . . .	11
6.4	The averages . . . . .	12
<b>7</b>	<b>Conclusion</b>	<b>14</b>
	<b>References</b>	<b>15</b>

## 1 Introduction

The overall vulnerability of the electrical grid appears to be increasing. This is because of different factors including the liberalization of the electrical market, the growth of demand and the appearance of new energy sources with a random production (solar and wind for instance). Today, engineers are undergoing research to try and enforce the reliability of the grid and propose various topological and technological solutions... One problem they have is that it's difficult to see if a configuration is better than another.

The goal of this project is to develop a tool that can evaluate the security and reliability of various power systems and its topologies. Using this tool, should make it possible to evaluate if a given power system topology of is better than another. Actually, tools that are able to do this task already exist [1]. However, they work statically by doing load flow iterations. The improvement of this program would be to do the simulation dynamically. The criterion used to define if a configuration is secure and stable will be the loss of load probability(LLP).

In order to evaluate this LLP, the first thing to do is to define all possible scenarios. Then, there are two ways to calculate:

1. The simulations of all scenarios are done. Then, the average loss of load is calculated while taking into account the scenario probability.
2. Only a part of the possible scenarios are chosen according to Monte Carlo Simulation (MCS) [2] [1]. The simulation is applied to the selected scenario. Then, the average loss of load can be calculated.

The first method is the most reliable to evaluate the loss of load, but it's not applicable to large grids because of the required time for simulation. Thus, in this work, the second method shall be applied .

This report will focus on some major points of the tools. First it will show how the program uses the MCS method and how scenarios are chosen. It shall be assumed that the number of events in a scenario follow a Poisson distribution. If few-event scenarios appear more often, they shall have smaller effects. On the contrary, several-event scenarios are rarer, but have a more important impact. Then, this document will focus on how the simulation is done, particularly on Eurostag and the API used. The report will show the rules applied to disconnect lines, machines and loads during simulation, if the limits of the system are not respected. After this, the report will show how to average the loss of load and how to calculate the probability density function (PDF). In order to classify the result, the PDF of the simulation can be approximated by a sum of Gaussian PDF. This is called the Gaussian mixture method (GMM) [2] [1]. ] In the end, simulation results will be exposed.

Because of the numerous hypothesis' done, the tools will not be able to give the exact loss of load probability (LLP). The main function of the program will be to compare two different topologies and decide which one is best.

### 1.1 Definitions

**Event:** An event is to be understood in the sense of Eurostag event. It is an action (voluntary or not) on the grid. The program will send events to Eurostag in order to open lines, shut down generators or disconnect loads.

**Scenario:** A scenario is defined as the starting point to a simulation. The system is in a steady state and one or more events happens.

## 2 General method

As the calculation of the loss of load probability is not directly possible on a grid, it is necessary to simulate different scenarios and analyse the results in order to obtain it. Because the probability of each scenario could be easily evaluated, one way to calculate the loss of load probability would be to simulate all scenarios. Unfortunately, due to the number of scenarios, it would take far too long. As a result, only some scenarios shall be simulated. To do so, we will use the Monte Carlo Method [2] [1].

### 2.1 The Monte Carlo simulations

The MCS method followed by the program is given by figure 1. The program will choose a random scenario to test, simulate it and save the result. After doing this N times, it will average the result in order to get the Loss of Load probability.

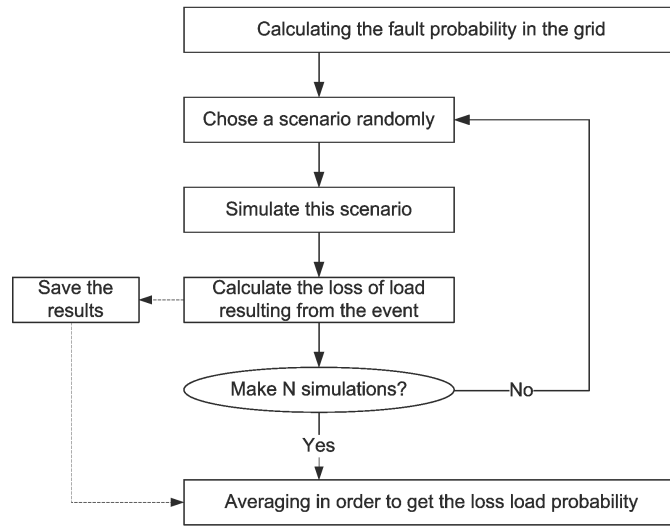


Figure 1: MCS Organogram

#### 2.1.1 List of all possible scenarios

In this case, two different events have been considered: line failures and generators breaking down. This means the total number of possible events is given by the relation:

$$N_{event-tot} = N_{machines} + N_{lines} \quad (1)$$

Considering that a scenario could have more than one event, the total number of scenarios is:

$$N_{scenarios} = \sum_{k=0}^n C_n^k = \sum_{k=0}^n \frac{n!}{k!(n-k)!} = 2^n \quad (2)$$

where  $n = N_{event-tot}$

This last number is huge for a grid with lots of components. Thus, not all the possible scenarios will be considered, but only the most likely.

Lets associate a probability to each event. For  $i = 1, 2, \dots, N_{event-tot}$ , the quantity  $pe_i$  is defined as the probability that the event  $i$  will happen in a steady state during a time  $T$  (These probabilities are defined in section 2.4). As two events will never happen exactly at the same time, there is a need to set a constant that defines the maximum elapsed time between two events considered as synchronous. This constant will be  $T$ . In this program  $T$  will be equal to 10 minutes:

$$T = 600[s]$$

### 2.1.2 Calculating the probability of scenarios

The equation (2) gives one very important piece of information. If the number of events is big (which is the case on big grids), then the number of scenarios is really huge. Because of this, it's impossible to calculate the probability of appearance of all scenarios, chose one and simulate it. Another assumption is needed.

**Hypothesis:** The events are uncorrelated and their number during a T time will follow a Poisson distribution.

## 2.2 The Poisson law

The Poisson law tells the probability that k events will happen in a given T time knowing that one event happens in average  $\lambda$  times during T. It is given by:

$$pp_k = e^{-\lambda} \cdot \frac{\lambda^k}{k!} \quad (3)$$

where  $pp_k$  is the probability that k events happen.

The figure number 2 shows Poisson distributions with various  $\lambda$ .

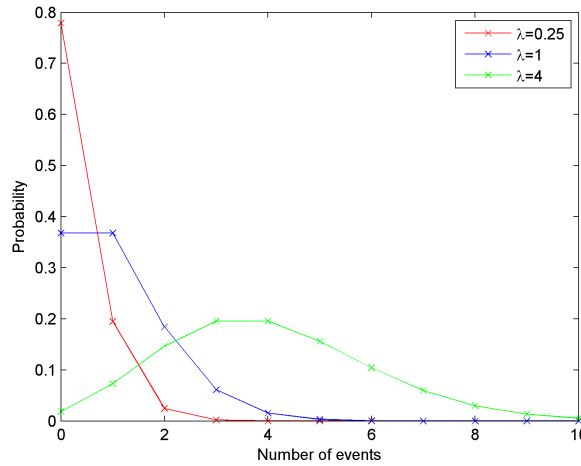


Figure 2: Poisson distributions with various  $\lambda$ .

Now, it is needed to determine  $\lambda$ . The average numbers of event appearances' are:  $pe_i$  with  $i = 1, 2, 3, \dots, N_{event}$  (because they are much smaller than one, the probability is equal to the appearance rate). In this case,  $\lambda$  is just equal to the sum of all  $pe_i$ .

**Note:** As  $\lambda$  is usually small, the probability that more than one event will happen is small. In order to test several-event scenarios more often,  $\lambda$  can be increased by a factor  $\alpha$  (This parameter could represent the influence of the weather for instance). Thus, in the program,  $\lambda$  is defined:

$$\lambda = \alpha \cdot \sum_{i=1}^{N_{event-tot}} pe_i \quad (4)$$

## 2.3 Chosing one scenario

There are two ways to select scenarios.

### 2.3.1 The logical strategy

The choice of one scenario is done in two steps. First, according to the Poisson distribution a number of events will be chosen. Secondly the events will be determined randomly according to their probability of appearance.

This technique is logical but not really efficient. In fact,  $\lambda$  is small, so several event scenarios will hardly ever be chosen. For this reason, we shall use another scenario strategy, that shall be developed further.

**Remark:** If the goal is only to compare two configurations, it's possible to increase  $\lambda$  with the same factor in both configurations. It will increase the number of events in scenarios and thus the loss of load. In this case, comparisons can be done with fewer simulations.

### 2.3.2 Another scenario strategy

In order to be more efficient and more precise, it is possible to adopt a scenario strategy that is a way to cover more scenarios with fewer simulations. As said earlier, few-event scenarios are unlikely to happen compared to several-event scenarios. As a result, by choosing random scenarios (according to their probability of appearance), they will hardly ever be simulated. The number of simulations needed in order to get several-event scenarios could be huge. In this case, the MCS is not so effective.

To avoid this problem, the strategy would be to simulate 1-event, 2-events, ..., l-events scenarios separately ( $N_s$  simulations each). Let's assume that we are considering scenarios with more than l events. In this report l will be written N-max-event.

The overall method is summarised in figure 3

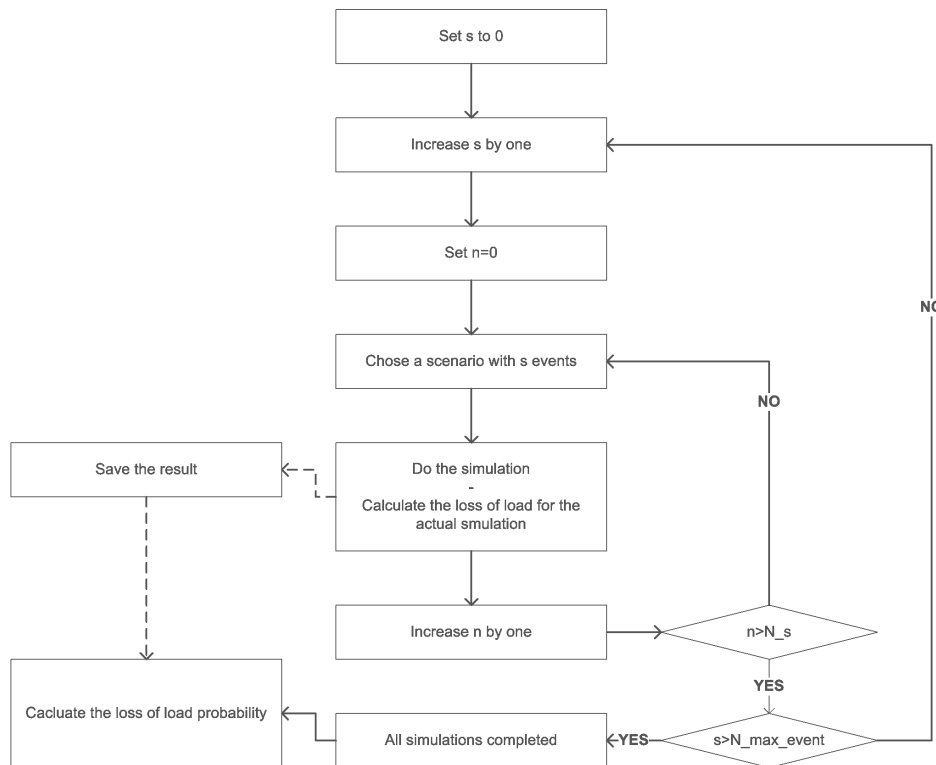


Figure 3: Overall method for an optimized scenario strategy.  $N$  is the number of simulations for a  $k$  events scenario.  $N_{\text{max-event}}$  is the maximum number of events simulated. The total number of simulations is  $N_s \cdot N_{\text{max-event}}$ .

**Another advantage:** This method also allows the user to adjust the parameter  $\alpha$  after the simulation (under certain conditions).

### 2.3.3 Correlation between events

Throughout this report, the assumption of non-correlated events in a scenario is done many times. This hypothesis is unfortunately not true. Therefore, the program includes an option which takes into account the correlation. If it is activated, all events in a scenario will be chosen in the same region of the grid. The idea behind this is that if events are correlated, they probably have the same cause. Thus they should be in the same region.

## 2.4 Probability of an event in a steady and unoverloaded state

In this part, in order to calculate the probability of an event, some hypothesis' have to be made.

1. Only failures of lines and generators are considered.
2. At beginning of the simulation, the system is in a steady state with no overload.
3. The probabilities between events are uncorrelated.
4. The probability of a generator breaking down is constant.
5. The probability of a line breaking down is:  $p = a * l + b$  where  $l$  is the length of the line in kilometres and  $a, b$  two constants.

### 2.4.1 The values used

In the following book [3], lots of values can be found. The chosen ones are these:

**Generators:** The outage rate of a generator for one year is  $p_g = 0.08$ .

**Lines:** The outage rate of a line(taking into account two transformer) for one year is  $p_l = 0.005 * l + 0.3$  (with  $l$  being the length of the line).

### 2.4.2 Finding the length of the lines

Unfortunately, the length of the lines is not accessible in Eurostag. Thus there is a need to estimate with the other line parameters.

In Eurostag, it is possible to have access to the nominal resistance of a line:  $R_{bn}$ . To convert it into real value, the following formula is used:

$$R_b = R_{bn} \cdot \frac{V_b^2}{S_b} \quad (5)$$

With  $V_b = 380kV$  and  $S_b = 100MVA$  (those values are defined in eurostag parameters).

The following assumptions are made: the length of the line is proportional to its resistance and all the lines are similar. Thus we have:  $R = l * r$ . In Kundurs book [4], the value  $r = 0.035[ohm/km]$  is given. The length (in kilometres) is given by:

$$l = \frac{R_{bn}}{r} \cdot \frac{V_b^2}{S_b} \quad (6)$$

### 2.4.3 Calculating the $pe_i$ values

In section 2.4, the outage rate for one year is given. In order to get the probability of failure for a T time (10 minutes), we have to apply this formula:

$$pe = p[1/year] \cdot \frac{600[s]}{31557600[s/year]} \quad (7)$$

**Note:** This is an approximation, but the result should not be too bad as  $p$  is not too big. The outage rate is nearly equal to the probability when it is far smaller than one.

### 3 The simulation

The simulation is done by the Eurostag program. As the interface does not allow the user to program several simulations with random initial points, an API is used in order to communicate with another program written in Python. An organogram is given in figure 4.

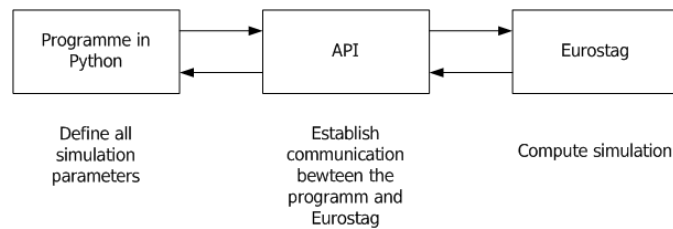


Figure 4: API organogram

The organogram of the simulation is given in figure 5. As the API does not allow to follow real time monitoring, the program will periodically check the state of the grid and send the event(s) to Eurostag (Disconnect lines, machines or loads). In order to do this, a time step is chosen. (It's not the same as in Eurostag.) During this time step no event can be sent. Thus, it should be smaller than the reaction time of the grid. But, in order to avoid unnecessary iteration of the program, it should not be too small either.

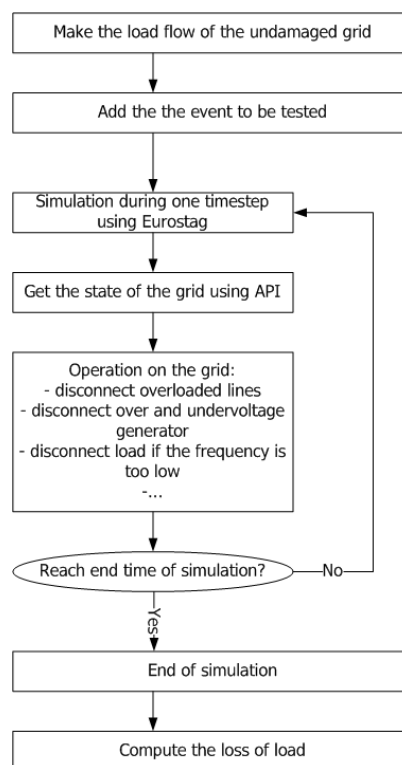


Figure 5: Simulation organigram



### 3.1 Testing the grid

The program will test three different things:

1. If lines are overloaded
2. If generator's nodes are over or under voltage
3. If the frequency is too big or too small

According to this test, the program is going to take a decision and will send the events to Eurostag.

#### 3.1.1 Lines

If the current is too big in a line, it has a certain probability of being disconnected. It is given in figure 6. Normally, the probability should not be zero when the line is not overloaded, but as the working time is short, it is assumed that random events will not happen.

**Hypothesis:** If the current (in p.u) is greater than 1.4, the line is always disconnected. If it's smaller than 1.1, the line is never disconnected. Between these two values, the following function has been used.

$$PCF_{line} = \frac{e^{(x-1.1)^\beta} - 1}{e^{(1.4-1.1)^\beta} - 1} \quad (8)$$

The parameter  $\beta$  is equal to 20, but can be adjusted taking to account T and real data.

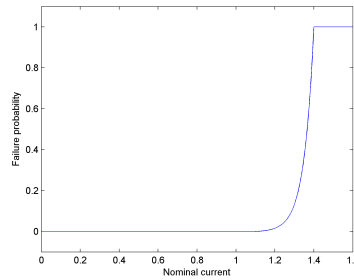


Figure 6: Probability of a line breaking down according to its nominal current for 1 s.

#### 3.1.2 The generators

A generator has to be stopped if the voltage at its node is either too small or too big (see figure 7), but also if it loses the synchronism (this is handled by the frequency control: section 3.2.1).

**Hypothesis:** If the voltage (in p.u) is greater than 1.2 or smaller than 0.8, the line is always disconnected. If it's between 0.9 and 1.1, the line is never disconnected. Between 0.8 and 0.9, respectively 1.1 and 1.2, these functions have been used.

$$PCF_{generator_m} = \frac{e^{(0.9-x)^{\beta_2}} - 1}{e^{(0.9-0.8)^{\beta_2}} - 1} \quad (9)$$

$$PCF_{generator_s} = \frac{e^{(x-1.1)^{\beta_2}} - 1}{e^{(1.2-1.1)^{\beta_2}} - 1} \quad (10)$$

The parameter  $\beta_2$  is equal to 20. This parameter can also be adjusted taking to account T and real data.

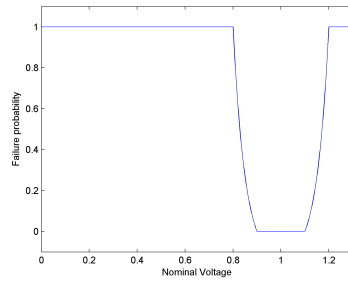


Figure 7: Probability of a generator breaking down according to its nominal voltage for 1 s.

### 3.2 End of simulation

A criterion for the end of the simulation has to be defined. There are two possibilities:

1. The simulation stops when the grid is in a steady state
2. The simulation stops after a certain time of simulation.

There are two main advantages in favour of the first stopping criterion. (a) If a steady state is quickly obtained, the simulation will stop right away and this will spare time. (b) If it takes a long time to get to the steady state, the right result will be obtained (maybe not with the second criterion).

Despite this, the second criterion will be used. One reason is, it's not easy to evaluate if the grid is in a steady state. Moreover, as Eurostag uses dynamic step sizes, if a steady state is obtained, the simulation will be fast.

The simulation time is going to be 3 minutes. This time has been chosen because it is more or less the human reaction time when there is a failure on the grid.

#### 3.2.1 Regulation strategy

The primary and secondary trimming is already implemented in every generator. However, it is expected to be insufficient in order to keep the grid in a viable situation. Thus, there will have to be load shedding if the frequency is too low. If it is too big, the generator will have to be shut down.

The limit values for the frequency are, for loads:  $f_{min} = 48.5Hz$  and  $f_{max} = 52Hz$  and for generators:  $f_{min} = 48Hz$  and  $f_{max} = 51.5Hz$ .

**Choice of the load to shed or the generator to shut down:** The load will be shed on the node of which the frequency is too low or too big.

**Note:** This approach, will also shut down generators which lose synchronism because their frequency will increase.

## 4 Calculation of the average loss of load

Once all simulations are done, it is possible to evaluate the loss of load probability. The calculations are different depending on the scenario strategy.

### 4.1 Traditional strategy

With the traditional MCS strategy (see section: 2.3.1), the loss of load is simply an average of all the previous results. The loss of load probability if a scenario is happening is:

$$PLL_{classic-fa} = \frac{1}{N} \cdot \sum_{i=1}^N LL_i \quad (11)$$

with  $N$  being the total number of simulations and  $LL_i$  the obtained loss of load.

## 4.2 Optimised strategy

With the optimised scenario strategy (see section: 2.3.2), the calculation is a little more tricky. It becomes weighted average. The loss of load probability if a scenario is happening is:

$$PLL_{optimised-fa} = \sum_{s=1}^{N_{max-event}} pp_s \cdot \left( \frac{1}{N_s} \cdot \sum_{i=1}^{N_s} LL_{si} \right) \quad (12)$$

with  $N$  being the number of k-event simulations and  $N_{max-event}$  the maximum number of events in a simulation.  $pp_s$  is the probability that  $s$  events happen given by the Poisson law.

## 4.3 Annualized expected loss of load

Now it's possible to annualize the loss of load probability. It is important to pay attention to the probability that no event shall happen. For a duration of  $T$  (600 s), the loss of load probability is given by:

$$PLL = pp_0 \cdot 0 + \left( \sum_{s=1}^{\infty} pp_s \right) PLL_{fa} = \left( \sum_{s=1}^{\infty} pp_s \right) PLL_{fa} \quad (13)$$

Then the expected loss of load per year is:

$$LL_{year} = PLL \cdot \frac{31557600[s/year]}{600[s]} \quad (14)$$

## 4.4 The probability density function (PDF)

It's possible to evaluate the PDF function of the loss of load.

As everything is discrete, the histogram for  $M$  different bins is given by the formula:

$$h(x) = \frac{1}{N \cdot h} \sum_{i=1}^N w_i \cdot k\left(\frac{x - x_i}{h}\right) \quad (15)$$

with  $h = \frac{\max(PLL) - \min(PLL)}{M}$ , and

$$k(t) = \begin{cases} 1/2 & \text{if } |t| \leq 1 \\ 0 & \text{if } |t| > 1 \end{cases}$$

and  $w_i$  the weight. They are all equal to 1 for the original scenario strategy and equal to  $pp_s \cdot N_{event-max}$  for the optimised scenario strategy (in optimised strategy  $N = N_s \cdot N_{event-max}$ ).

## 5 Gaussian mixture method (GMM)

The MCS will return an array of Loss of Load. To get an approximation of the PDF, the easiest way would be to plot a normalized histogram of the obtained values. This is equivalent to doing a linear approximation of the PDF with data from an experimental realisation.

In order to analyse the results in a more detailed manner, a GMM is applied. The idea is to approximate the PDF by a sum of weighted Gaussian PDFs. This is not the only possible method. Piecewise linear estimation and exponential estimation are also possible. However GMM seems to be the most reliable [2] [1].

To approximate a PDF, two functions are needed: a kernel and a window.

**The kernel:** The kernel is, as expected a normalized Gaussian with a mean  $\mu$  and a variance  $\sigma^2$ . It is given by the formula:

$$N(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x - \mu)^2}{2\sigma^2}\right] \quad (16)$$

**The window:** The Parzen window is chosen because it doesn't make any assumptions on the form of the PDF. The shape is entirely determined by the data without having to choose the location of the centres. With this windows, the PDF is given by:

$$p(x; \theta) = \sum_{c=1}^C \alpha_c N(x; \mu_c, \sigma_c^2) \quad (17)$$

Where the  $\alpha_c$  being the weight of all components with  $\sum_{c=1}^C \alpha_c = 1$ . All  $\alpha_c$  are  $w_i/N$  (with  $N$  equal to the total number of simulations).  $\sigma$  is taken to 0.01 to smooth the PDF. The  $\mu_c$  are the results of the MCS.

This defines the set of parameters for a particular PDF Gaussian mixture.

$$\theta = \{\alpha_1, \mu_1, \sigma, \dots, \alpha_c, \mu_c, \sigma\} \quad (18)$$

An example is shown in figure 8

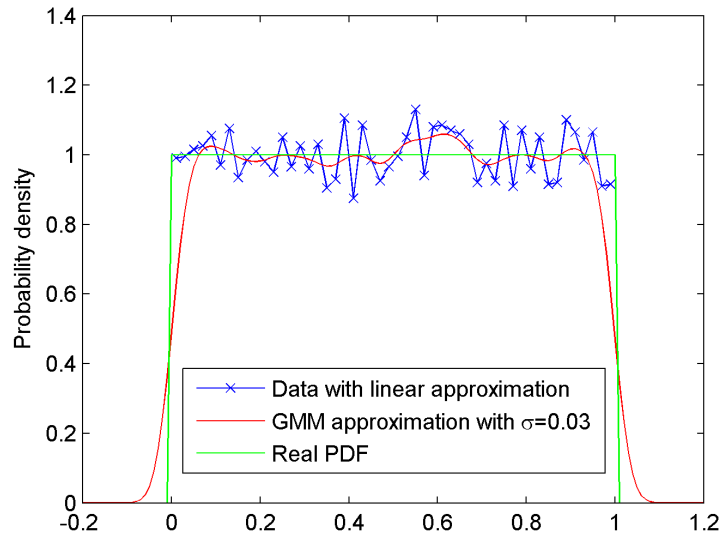


Figure 8: GMM applied to a uniform distribution with 10'000 samples. The blue curve is the histogram of data with 50 bins (given by (15)). The red curve is the GMM approximation with  $\sigma = 0.03$  (given by (17)).

## 6 Results

In this section, results shall be presented and discussed. The grid used for simulations is a simplified model of the European and Russian very high voltage grid presented in [5]. In this section, the goal is to test the program. To observe if everything is correct, high consequence scenarios will be simulated. In order to achieve this task, the parameter  $\alpha$  is set to 5000 and the total load of the grid (929 GW) is equivalent to 1 pu. As a consequence, there are in average 8 initial events in a scenario. Three tests shall be done. First the traditional and the optimised strategy will be compared. Secondly, the effect of correlation will be tested. Thirdly, the influence of the parameter  $\alpha$  will be shown. In the end, the averages of loss of load will be presented.

### 6.1 Comparison between traditional and optimised strategy:

This comparison is shown in figure 9. The two curves have a similar shape but they do not fit exactly. Since MCS are random, this result was expected. At this point, it is impossible to decide which curve gives the best result. For convenience, the optimized strategy will be used later on.

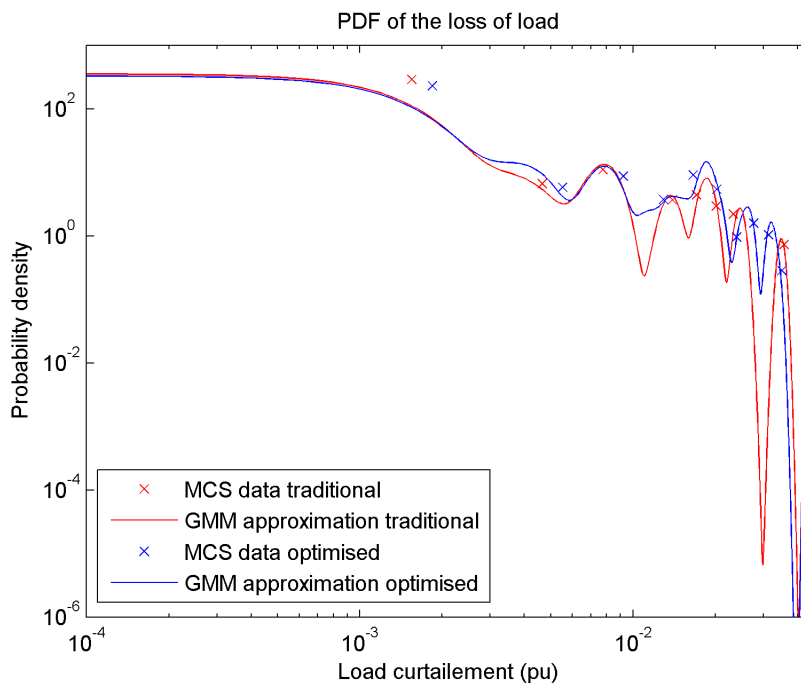


Figure 9: Comparison between traditional and optimised strategy for 500 simulations and uncorrelated scenarios.  $\alpha = 5000$ ,  $\sigma = 0.001$ .

### 6.2 Comparison between correlated and uncorrelated scenarios:

As shown in figure 10, the curves have similar shapes. However, there is a big difference between them. Correlated scenarios sometimes have far bigger effects than uncorrelated ones. In figure 10, the biggest load curtailment obtained is 0.05 pu for uncorrelated and 0.6 pu for correlated.

### 6.3 Influence of parameter $\alpha$ :

$\alpha$  is used to increase  $\lambda$ , the Poisson parameter. This last number represents the average number of events in a scenario. As a consequence, if  $\alpha$  is big, the loss of load will be bigger. The figure 11 confirms this idea.

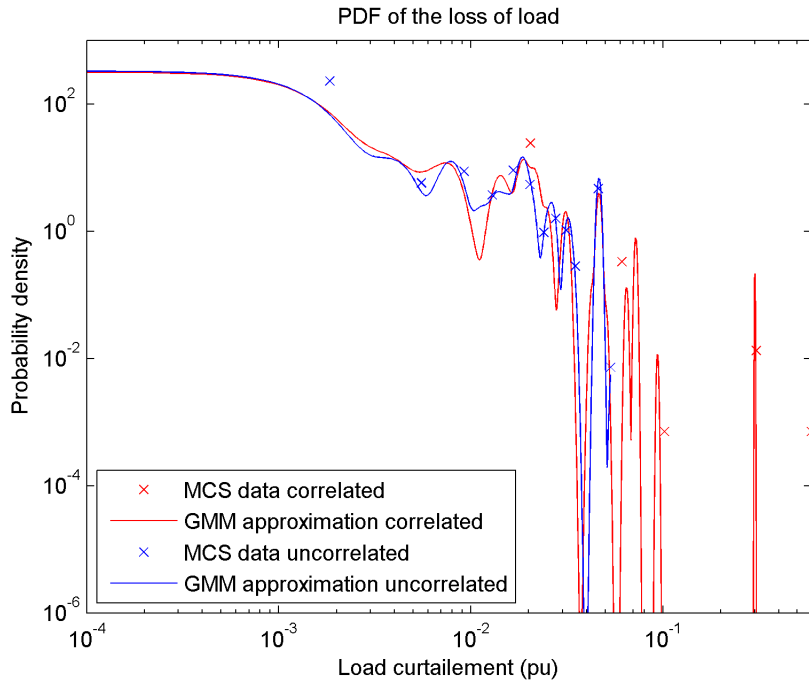


Figure 10: Comparison between correlated and uncorrelated scenarios for 500 simulations.  $\alpha = 5000$ ,  $\sigma = 0.001$ .

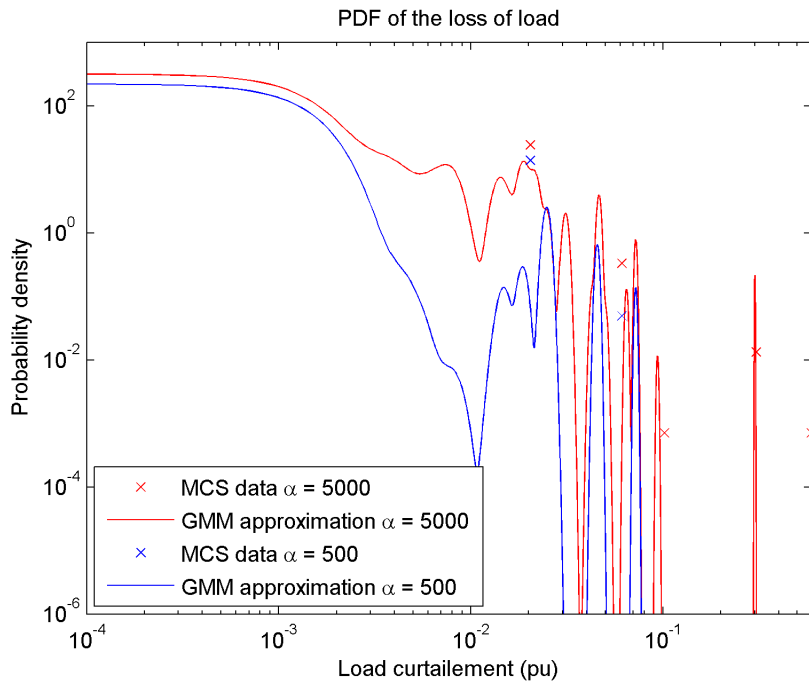


Figure 11: Influence of  $\alpha$  on the result. Correlated scenario,  $\sigma = 0.001$ .

#### 6.4 The averages

The averages of loss of load probability are given in table 1. This result was not expected. (Optimised and traditional strategy should have the same values). However, MCS are a random process. More simulations should be done to establish conclusions.

	Traditional strategy	Optimised strategy
<b>Uncorrelated</b>	0.0014	0.0027
<b>Correlated</b>	0.0042	0.0032

Table 1: Result: average loss of load if a scenario happens ( $PLL_{fa}$ ). 500 simulations with  $\alpha = 5000$  have been done.

**Effect of  $\alpha$ :** The effect of  $\alpha$  can be observed in table 2.

$\alpha$	$\lambda$	$PLL_{fa}$
5000	8.2795	0.0032
2500	4.1398	0.0025
1250	2.0699	0.0013
500	0.82795	0.00027

Table 2: Result: average loss of load if a scenario happens ( $PLL_{fa}$ ). 500 simulations with various  $\alpha$  have been done. The scenarios are correlated.

**Annual loss of load average:** The annual loss of load of the grid can be computed with 13 and 14.  $\alpha$  is equal to 1 for uncorrelated scenarios and equal to 50 for correlated scenarios. The calculation is done with an optimized strategy with only 1,2,3,4,5-event scenario simulated.

The average loss of load for uncorrelated scenarios is **0.0145** pu which corresponds to 13'507 MW.

The average loss of load for correlated scenarios is **0.2808** pu which corresponds to 260'930 MW.

**A last simulation:** In order to get a LLP close to reality,  $\alpha$  is equal to 1. Only few events scenarios are done. For this example, 300 single event, 200 double-event and 100 3,4,5-event simulations have been done. The result is shown on figure 12.

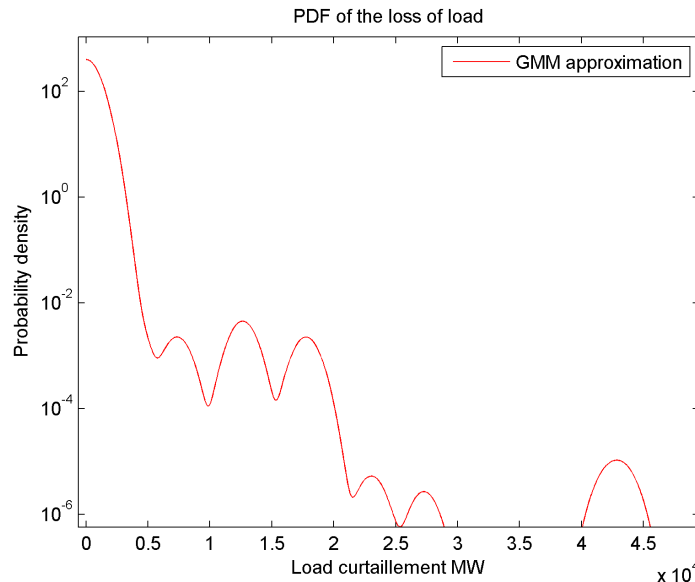


Figure 12: Loss of load probability density function for ten minutes. The average is 0.3671 MW (19'309 MW per year). Correlated scenarios,  $\alpha = 1$ ,  $\sigma = 0.001$ .

## 7 Conclusion

In this work, a tool has been developed in order to compare different grid' topologies. As a direct calculation is not possible, MCS have been used. In order to speed up the simulation an optimised method has been developed. In MCS, scenarios are chosen according to their probability of appearance. Thus multi-event scenarios are rarely simulated, but they are important due to their big effect. The optimised strategy can force the simulation of multi event simulations. Moreover, it allows the user to adjust the parameter  $\alpha$  after the simulation which allows the user to fit the curve with real data. In order to be able to calculate the probability of a scenario, lots of hypothesis have been done. The most important is that the number of events in a scenario follow a Poisson law.

The program has been tested on a simplified European and Russian grid [5]. As the comparison with real data has not been done yet, it is not possible to assure that the results are correct. However, the curves looks good and the results are plausible. The traditional and the optimised strategies give similar PDFs. Thus, the optimal strategy should affect negatively the result. Moreover, the correlation seems to encourage big failure on the grid.

One problem of the program is the time needed to simulate. As all simulations are made dynamically, the results should improve compare to statics methods (load flow iterations [1]). However, in order to get a precise approximation, lots of simulations have to be done. So, to verify the results and to check the speed of convergence of the program, a comparison with a static tool would be good. The program can also be improved with a graphical interface or with more precise data. Lines and generators probabilities are available, so they could be added directly into the program.

The constants (like  $\alpha$ ) of the program or the scenario model (Poisson law) need to be adjusted to fit reality. However, this task is not trivial as statistics about correlation in failure are not available.



## References

- [1] O. Mousavi, G. Gharehpetian, and M. Naderi, “Estimating risk of cascading blackout using probabilistic methods,” in *Electric Power and Energy Conversion Systems, 2009. EPECS '09. International Conference on*, nov. 2009, pp. 1–4.
- [2] O. A. Mousavi, M. S. Farashbashi-Astaneh, and G. B. Gharhpetian, “Improving power system risk evaluation method using monte carlo simulation and gaussian mixture method,” *Advances in Electrical and Computer Engineering*, vol. 9, pp. 38–44, 2009.
- [3] C. Grigg, P. Wong, P. Albrecht, R. Allan, M. Bhavaraju, R. Billinton, Q. Chen, C. Fong, S. Haddad, S. Kuruganty, W. Li, R. Mukerji, D. Patton, N. Rau, D. Reppen, A. Schneider, M. Shahidehpour, and C. Singh, “The ieee reliability test system-1996. a report prepared by the reliability test system task force of the application of probability methods subcommittee,” *Power Systems, IEEE Transactions on*, vol. 14, no. 3, pp. 1010–1020, aug 1999.
- [4] P. Kundur, *Power system stability and control*, N. J. Balu and M. G. Lauby, Eds. McGraw-Hill, 1994.
- [5] A. G. e. a. A. Arestova, U. Häger, “Supersmart grid for improving system stability at the example of a possible interconnection of entso-e and ips/ups,” vol. 9, 2011.